CLAIMS

1. A method of performing numerical computations in a mathematical system comprising at least one function, the method comprising the steps of:

- expressing the mathematical system in discrete terms,
- expressing at least one variable of the mathematical system as a fixed-point number,
- performing said computations in such a way that the computations include the at least one variable expressed as a fixed-point number,
- obtaining, from said computations, a resulting number, the resulting number representing at least one of:
  - a. at least a part of a solution to the mathematical system, and
  - b. a number usable in further computations involved in the numerical solution of the mathematical system,

the method further comprising:

- extracting a set of data which represents at least one of:
  - i. a subset of digits of the resulting number, and
  - ii. a subset of digits of a number derived from the resulting number.

2. A method according to claim 1, wherein said set of data represent a pseudo-random number.

3. A method according to claim 1, wherein said computations involve at least a first and a second fixed-point number, each fixed-point number having a decimal separator, wherein the decimal separator of the first fixed-point number is positioned at a position different from the position of the decimal separator of the second fixed-point number.

4. A method according to claim 3, wherein the step of performing computations involves positioning the decimal separator of the first and second fixed-point number at selected positions.

5. A method according to claim 1, wherein said at least one function is non-linear.

6. A method according to claim 1, wherein the resulting number is expressed as a variable selected from the group consisting of:

- an integer number,
- a floating point number, and
- a fixed-point number.

7. A method according to claim 1, wherein the mathematical system comprises at least one of:

- a differential equation,
- a discrete mapping.

8. A method according to claim 7, wherein the differential equation comprises at least one of:

- a partial differential equation,
- an ordinary differential equation.

9. A method according to claim 7, wherein the discrete mapping comprises at least one of:

- an area-preserving map,
- a non area-preserving map.

10. A method according to claim 7, wherein the mathematical system comprises at least one non-linear function governing at least one state variable X.

11. A method according to claim 10, wherein the mathematical system comprises a set of non-linear mapping functions.

12. A method according to claim 9, wherein the map comprises at least one of:

- a logistic map of the form:
  $x_{n+1}=\mu x_n(1-x_n)$,
- an Anosov map of the form:
  $$\begin{bmatrix} x_{n+1} \\ y_{n+1} \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ 1 & 2 \end{bmatrix} \begin{bmatrix} x_n \\ y_n \end{bmatrix} \bmod 1,$$
- a Hénon map of the form:
  $$\begin{bmatrix} x_{n+1} \\ y_{n+1} \end{bmatrix} = \begin{bmatrix} 1 + y_n - a x_n^2 \\ b x_n \end{bmatrix}.$$

13. A method according to claim 1, wherein the mathematical system comprises at least one non-linear differential equation.

14. A method according to claim 13, wherein the mathematical system comprises a set of non-linear differential equations.

15. A method according to claim 7, wherein the mathematical system has at least one positive Lyapunov exponent.

16. A method according to claim 7, comprising computing at least one Lyapunov exponent at least once during the mathematical computations.

17. A method according to claim 13, wherein the at least non-linear differential equation governs at least one state variable, X, which is a function of at least one independent variable, t.

18. A method according to claim 14, wherein the set of non-linear differential equations is a Lorenz system.

19. A method according to claim 18, wherein the Lorenz system consists of the following differential equations:

$$\frac{dx}{dt} = \sigma(y - x)$$
$$\frac{dy}{dt} = rx - y - xz,$$
$$\frac{dz}{dt} = xy - bz,$$

wherein X=(x, y, z) are state variables, t is the independent variable, and σ, r and b are parameters.

20. A method according to claim 13, wherein the step of performing computations comprises numerically integrating at least one of:

- the non-linear differential equation, and
- the non-linear differential equations of said set of non-linear differential equations,

by repeatedly computing a solution X_{n+1} based on at least one previous solution X_m, $m \le n+1$, and a step length, ΔTn, of the independent variable, t.

21. A method according to claim 20, wherein the step of integrating comprises providing at least one initial condition, X_0, of the state variable, X, and an initial step length, ΔT_0.

22. A method according to claim 10, wherein the step of performing computations comprises numerically iterating the non-linear mapping function.

23. A method according to claim 22, wherein the step of iterating comprises providing at least one initial condition, X_0, of the state variable, X.

24. A method according to claim 20, wherein, in the discretized formulation of the Lorenz system, the solution X_{n+1} is computed using the step length $\Delta T = (\Delta t_{x,n}, \Delta t_{y,n}, \Delta t_{z,n})$ as follows:

$$x_{n+1} = x_n + (\sigma(y_n - x_n)) \cdot \Delta t_{x,n}$$
$$y_{n+1} = y_n + (x_n(r - z_n) - y_n) \cdot \Delta t_{y.n}$$
$$z_{n+1} = z_n + (x_n y_n - bz_n) \cdot \Delta t_{z,n},$$

wherein:

$\Delta t_{x,n}$ is the step length used in the computation of x_{n+1},
$\Delta t_{y,n}$ is the step length used in the computation of y_{n+1},
$\Delta t_{z,n}$ is the step length used in the computation of z_{n+1}.

25. A method according to claim 20, wherein the step length ΔT is constant throughout the computations.

26. A method according to claim 20, wherein, in each integration step, at least one of the elements $(\Delta t_{x,n}, \Delta t_{y,n}, \Delta t_{z,n})$ of the step length ΔT is a function of at least one number related to said computations.

27. A method according to claim 26, wherein, in each integration step, at least one of the elements ($\Delta t_{x,n}$, $\Delta t_{y,n}$, $\Delta t_{z,n}$) of the step length ΔT is a function of at least one solution, X_m, which is a solution to the mathematical system.

28. A method according to claim 26 wherein, in each integration step, at least one of the elements ($\Delta t_{x,n}$, $\Delta t_{y,n}$, $\Delta t_{z,n}$) of the step length ΔT is a function of at least one given step length, ΔT_m.

29. A method according to claim 1, wherein a key selected from an encryption key and a decryption key is used to determine at least one value of at least one variable in the mathematical system.

30. A method according to claim 29, wherein the key is used to determine at least a part of the initial condition X_0.

31. A method according to claim 29, wherein the key is used to determine at least a part of the initial step length ΔT_0.

32. A method according to claim 29, wherein the key is used to determine the at least a part of at least one of the parameters.

33. A method according to claim 29, wherein the key is a public key.

34. A method according to claim 29, wherein the key is a private key.

35. A method according to claim 1, comprising extracting a plurality of numbers resulting from the computations.

36. A method according to claim 1, wherein the step of extracting comprises extracting at least one number derived from k bits of the resulting number.

37. A method according to claim 1, wherein the step of extracting comprises extracting the k least significant bits of the resulting number.

38. A method according to claim 36, wherein k is a value selected from the group consisting of: 8, 16, 32, 64, and 128.

39. A method according to claim 36, wherein a plurality of numbers are extracted.

40. A method according to claim 1, wherein the extracted set of data is manipulated by means of at least one of:
- an arithmetic operation, and
- a logical operation,

so as to obtain a combined set of data.

41. A method according to claim 40, wherein at least one of the:
- extracted set of data, and
- the combined set of data

is combined with original data, so as to encrypt the original data.

42. A method according to claim 40, wherein at least one of:
- extracted set of data, and
- the combined set of data

is combined with encrypted data, so as to decrypt the encrypted data and obtain the original data.

43. A method according to claim 40, wherein the combining of data comprises an XOR operation.

44. A method according to claim 1, wherein said computations involve data representing a block of plaintext in a block-cipher encryption and decryption system.

45. A method according to claim 1, wherein the extracted set of data is used to define at least one operation on a block of plaintext in a block-cipher encryption and decryption system.

46. A method according to claim 41, wherein the combining of data comprises addition of the original data and the combined set of data for encryption, and subtraction of the combined set of data from the encrypted data for decryption.

47. A method according to claim 41, wherein the combining of data comprises subtraction of the combined set of data from the original data for encryption, and addition of the combined set of data and the encrypted data for decryption.

48. A method according to claim 1, wherein the extracted set of data is used as at least one of: an encryption key and a decryption key.

49. A method according to claim 1, wherein the extracted set of data is used to generate at least one of: an encryption key and a decryption key.

50. A method according to claim 1, wherein the extracted set of data is used in generation of data representing a digital signature.

51. A method according to claim 1, wherein the extracted set of data is used in watermarking of digital data.

52. A method according to claim 1, wherein the computations are performed on an electronic device which comprises an electronic processing unit having a register width, the method comprising the steps of:

- expressing at least one integer number of a bit width larger than said register width as at least two sub-numbers each having a bit width which is at most equal to said register width,
- performing at least one of said computations as a sub-computation on each of the sub-numbers so as to arrive at at least two partial results, expressed as integer numbers of a bit width smaller which is at most equal to the register width of the processing unit,
- concatenating the partial results to yield a representation of a result of said at least one computation.

53. A computer program for performing numerical computations in a mathematical system comprising at least one function, the computer program being adapted to:

- express at least one variable of the mathematical system as a fixed-point number,
- perform said computations in such a way that the computations include the at least one variable expressed as a fixed-point number,
- obtain, from said computations, a resulting number, the resulting number representing at least one of:
  - a. at least a part of a solution to the mathematical system, and
  - b. a number usable in further computations involved in the numerical solution of the mathematical system,

the computer program being further adapted to:

- extract a set of data which represents at least one of:
  - i. a subset of digits of the resulting number, and
  - ii. a subset of digits of a number derived from the resulting number.

54. A computer readable data carrier loaded with a computer program for performing numerical computations in a mathematical system comprising at least one function, the computer program being adapted to:

- express at least one variable of the mathematical system as a fixed-point number,
- perform said computations in such a way that the computations include the at least one variable expressed as a fixed-point number,
- obtain, from said computations, a resulting number, the resulting number representing at least one of:
  - a. at least a part of a solution to the mathematical system, and
  - b. a number usable in further computations involved in the numerical solution of the mathematical system,

the computer program being further adapted to:

- extract a set of data which represents at least one of:
  - i. a subset of digits of the resulting number, and
  - ii. a subset of digits of a number derived from the resulting number.

55. A computer which is operatively connected to a computer readable data carrier loaded with a computer program for performing numerical computations in a mathematical system comprising at least one function, the computer program being adapted to:

- express at least one variable of the mathematical system as a fixed-point number,
- perform said computations in such a way that the computations include the at least one variable expressed as a fixed-point number,
- obtain, from said computations, a resulting number, the resulting number representing at least one of:
  - a. at least a part of a solution to the mathematical system, and
  - b. a number usable in further computations involved in the numerical solution of the mathematical system,

the computer program being further adapted to:

- extract a set of data which represents at least one of:
  - i. a subset of digits of the resulting number, and
  - ii. a subset of digits of a number derived from the resulting number,

wherein the computer comprises processor means for running said program.

56. A signal comprising an extracted set of data which have been derived from computations in a mathematical system, wherein, in order to arrive at the extracted set of data:

- the mathematical system has been expressed in discrete terms,
- at least one variable of the mathematical system has been expressed as a fixed-point number,
- said computations have been performed in such a way that the computations have included the at least one variable expressed as a fixed-point number,
- a resulting number has been obtained from said computations, the resulting number representing at least one of:
  - a. at least a part of a solution to the mathematical system, and
  - b. a number usable in further computations involved in the numerical solution of the mathematical system,

whereby the extracted set of data represents at least one of:

- i. a subset of digits of the resulting number, and
- ii. a subset of digits of a number derived from the resulting number.

57. A signal comprising an encrypted set of data which has been derived as a combination of plaintext and at least one set of data extracted from computations in a mathematical system, wherein, in order to arrive at the extracted set of data:

- the mathematical system has been expressed in discrete terms,
- at least one variable of the mathematical system has been expressed as a fixed-point number,
- said computations have been performed in such a way that the computations have included the at least one variable expressed as a fixed-point number,
- a resulting number has been obtained from said computations, the resulting number representing at least one of:

- a. at least a part of a solution to the mathematical system, and
- b. a number usable in further computations involved in the numerical solution of the mathematical system,

whereby the extracted set of data represents at least one of:

- i. a subset of digits of the resulting number, and
- ii. a subset of digits of a number derived from the resulting number.

58. A method of detecting periodic behavior in the solution of a mathematical system comprising at least one non-linear function governing at least one state variable with respect to at least one independent variable, the method comprising:

- expressing the mathematical system in discrete terms,
- performing computations in an electronic device so as to obtain resulting numbers, the resulting numbers representing at least parts of solutions to the mathematical system,
- storing selected solutions in an array, A, in a memory of the electronic device, the array being adapted to store a finite number, n+1, of solutions,
- determining whether at least one of:
  - a current solution, and
  - a particular one of said solutions stored in the array

  is substantially identical to another solution stored in the array.

59. A method according to claim 58, wherein only selected solutions are stored in the memory.

60. A method according to claim 58, wherein each entry in said array contains a solution having an age which is growing by array level, A_i, $0 \leq i \leq n$, the method comprising:

- at the step of storing selected solutions in the array: storing a current solution at the 0'th level in the array, A, thereby overwriting an old value stored at the 0'th level in the array, A,
- if a 0'th predetermined criterion is fulfilled: transferring the old value to the 1'st level in the array, A, before the 0'th level is overwritten by the current solution, and

for the 1st level and each further level i in the array:

- if an i'th predetermined criterion for level i is fulfilled: transferring the old value stored at the i'th level to the i+1'st level in the array, A, before the i'th level is overwritten by the value transferred from the i-1st level,

if the n'th level is to be updated: discarding the old value previously stored at the n'th level.

61. A method according to claim 60, further comprising, for each level, i, in the array, counting the number of times an old value stored at the i'th level has been overwritten by a new value without the old value being transferred to the i+1'st level, the i'th predetermined criterion being fulfilled if the old value has not been transferred for a predetermined number of times.

62. A method according to claim 61, wherein the predetermined number of times is the same for all levels of the array, A.

63. A method according to claim 61, wherein the predetermined number of times varies between the levels of the array, A.

64. A method according to claim 61, wherein the predetermined number of times for the i'th level of the array, A, is dependent from at least one value stored in the array.

65. A method according to claim 58, wherein said step of determining is only performed when a test criterion is fulfilled.

66. A method according to claim 65, wherein the test criterion is fulfilled when the sign of at least one state variable changes.

67. A method according to claim 66, further comprising computing at least one derivative of at least one state variable with respect to one of said at least one independent variable, the test criterion being fulfilled when there occurs a change of sign of said at least one derivative.

68. A method according to claim 66, further comprising computing a test value from at least one of:
- said at least one state variable, and
- said derivative,

the test criterion being based on the test value.

69. A method of generating a pseudo-random number, the method comprising:
I) expressing a mathematical system in discrete terms,
II) defining a seed value representing at least an initial condition for the mathematical system,
III) expressing at least one variable of the mathematical system as a fixed-point number,
IV) performing computations in an electronic device, the computations including the at least one variable expressed as a fixed-point number and obtaining, from said computations, a resulting number, the resulting number representing at least one of:
a. at least a part of a solution to the mathematical system, and
b. a number usable in further computations involved in the numerical solution of the mathematical system,
V) extracting, as the pseudo-random number, a number derived from at least one number which has occurred during the computations.

70. A method according to claim 69, wherein the pseudo-random number is extracted as a number derived from k digits of said at least one number which has occurred during the computations.

71. A method according to claim 70, wherein the pseudo-random number is extracted as a number derived from the k least significant digits of said at least one number.

72. A method according to claim 69, the method comprising the steps of repeating steps IV) and V) until a given amount of pseudo-random numbers has been generated.

73. A method according to claim 69, wherein a given amount of pseudo-random numbers is generated and stored in a memory of the electronic device as a spare seed value.

74. A method according to claim 69, wherein a plurality of resulting numbers are obtained which represent at least parts of solutions to the mathematical system, the method further comprising detecting periodic behavior in the solution of the mathematical system, the mathematical system comprising at least one non-linear function governing at least one state variable with respect to at least one independent variable, the detecting of periodic behavior comprising:

- storing selected solutions in an array, A, in a memory of the electronic device, the array being adapted to store a finite number, n+1, of solutions,
- determining whether at least one of:
  - a current solution, and
  - a particular one of said solutions stored in the array

  is substantially identical to another solution stored in the array,

the method further comprising:
if the step of determining reveals that at least one of
- the current solution, and
- the particular solution

is identical to another solution:
interrupting the pseudo-random-number generation, i.e. interrupting repetition of steps IV) and V),
using the spare seed value as the seed value in the step II),
resuming the pseudo-random-number generation, i.e. resuming repetition of steps IV) and V).

75. A method according to claim 74, further comprising, prior to the step of resuming the pseudo-random number generation, generating and storing, in a memory of the electronic device, a given amount of pseudo-random numbers as a new spare seed value.

76. A method according to claim 69, wherein each level in the array, A, is reset prior to step IV), when steps IV) and V) are initiated with a new seed value at step II).

77. A method of encrypting a set of original data into a set of encrypted data, the method comprising the steps of:
A) generating a pseudo-random number by performing the steps of:
I) expressing a mathematical system in discrete terms,

II) defining an encryption key representing at least an initial condition for the mathematical system,

III) expressing at least one variable of the mathematical system as a fixed-point number,

IV) performing computations including the at least one variable expressed as a fixed-point number and obtaining, from the computations, a resulting number, the resulting number representing at least one of:

a. at least a part of a solution to the mathematical system, and

b. a number usable in further computations involved in the numerical solution of the mathematical system,

V) extracting, as the pseudo-random number, a number derived from at least one number which has occurred during the computations,

B) manipulating the original data and the pseudo-random number by means of at least one of:

i. an arithmetic operation, and

ii. a logical operation,

so as to obtain a combined set of data, the combined set of data being the encrypted data.

78. A method according to claim 77, wherein, prior to step A), a sub-set of the original data is separated from the set of data, and wherein step B) is performed on the sub-set of data.

79. A method according to claim 77, wherein the pseudo-random number is extracted as a number derived from k digits of said at least one number which has occurred during the computations.

80. A method according to claim 77, wherein the pseudo-random number is extracted as a number derived from the k least significant digits of said at least one number which has occurred during the computations.

81. A method according to claim 77, the method comprising the steps of repeating steps IV) and V) until a given amount of pseudo-random numbers has been generated.

82. A method according to claim 77, wherein a given amount of pseudo-random numbers is generated and stored in a memory of the electronic device as a spare encryption key.

83. A method according to claim 82, wherein a plurality of resulting numbers are obtained which represent at least parts of solutions to the mathematical system, the method further comprising detecting periodic behavior in the solution of the mathematical system, the mathematical system comprising at least one non-linear function governing at least one state variable with respect to at least one independent variable, the detecting of periodic behavior comprising:

– storing selected solutions in an array, A, in a memory of the electronic device, the array being adapted to store a finite number, n+1, of solutions,

- determining whether at least one of:
  - a current solution, and
  - a particular one of said solutions stored in the array

  is substantially identical to another solution stored in the array,

the method further comprising:
if the step of determining reveals that at least one of:

- the current solution, and
- the particular solution

is identical to another solution:

- interrupting the pseudo-random number generation, i.e. interrupting repetition of steps IV) and V),
- using the spare encryption key as the encryption key in step II),
- resuming the pseudo-random number generation, i.e. resuming repetition of steps IV) and V).

84. A method according to claim 83, further comprising, prior to the step of resuming the pseudo-random number generation, generating and storing, in a memory of the electronic device, a given amount of pseudo-random numbers as a new spare encryption key.

85. A method according to claim 77, wherein each level in the array, A, is reset prior to step IV), when steps IV) and V) are initiated with a new seed value at step II).

86. A method of decrypting a set of encrypted data which has been encrypted by a method of encrypting a set of original data into said set of encrypted data, the method of encrypting comprising the steps of:
A) generating a pseudo-random number by performing the steps of:

I) expressing a mathematical system in discrete terms,
II) defining an encryption key representing at least an initial condition for the mathematical system,
III) expressing at least one variable of the mathematical system as a fixed-point number,
IV) performing computations including the at least one variable expressed as a fixed-point number and obtaining, from the computations, a resulting number, the resulting number representing at least one of:
a. at least a part of a solution to the mathematical system, and
b. a number usable in further computations involved in the numerical solution of the mathematical system,
V) extracting, as the pseudo-random number, a number derived from at least one number which has occurred during the computations,

B) manipulating the original data and the pseudo-random number by means of at least one of:
i. an arithmetic operation, and
ii. a logical operation,
so as to obtain a combined set of data, the combined set of data being the encrypted data,

the method of decrypting comprising the steps of:

a) performing step A), so as to extract the same pseudo-random number as extracted in step V),

b) manipulating the encrypted data and the pseudo-random number by means of at least one of:

- an arithmetic operation, and
- a logical operation,

so as to obtain the original, i.e. decrypted, version of the data.

87. A method according to claim 86, wherein, prior to step a), a sub-set of the encrypted data is separated from the set of encrypted data, the method of decrypting comprising performing steps a) and b) on said sub-set of data.

88. A method according to claim 87 comprising repeating the steps of claim 87 until a plurality of sub-sets which in common constitute the entire set of encrypted data have been decrypted.

89. A computer program for encrypting and decrypting a set of data, the computer program being adapted to run in an encryption mode and in a decryption mode, the computer program being further adapted to:

i) generate a pseudo-random number in a reproducible way by performing the steps of:

- expressing a mathematical system in discrete terms,
- expressing at least one variable of the mathematical system as a fixed-point number,
- performing computations including the at least one variable expressed as a fixed-point number,
- obtaining, from the computations, a resulting number, the resulting number representing at least one of:
  a. a part of a solution to the mathematical system, and
  b. a number usable in further computations involved in the numerical solution of the mathematical system,
- extracting, as the pseudo-random number, a number derived from at least one number which has occurred during the computations,

ii) manipulate the data and the pseudo-random number by means of at least one of:

- an arithmetic operation, and
- a logical operation,

so as to obtain a combined set of data, wherein:

- the combined set of data represents an encrypted version of the data in case the computer program is run in encryption mode,
- the combined set of data represents a decrypted version of the data in case the computer program is run in decryption mode.

90. A computer readable data carrier loaded with a computer program for encrypting and decrypting a set of data, the computer program being adapted to run in an encryption mode and in a decryption mode, the computer program being further adapted to:

i) generate a pseudo-random number in a reproducible way by performing the steps of:

- expressing a mathematical system in discrete terms,
- expressing at least one variable of the mathematical system as a fixed-point number,
- performing computations including the at least one variable expressed as a fixed-point number,
- obtaining, from the computations, a resulting number, the resulting number representing at least one of:
  a. a part of a solution to the mathematical system, and
  b. a number usable in further computations involved in the numerical solution of the mathematical system,
- extracting, as the pseudo-random number, a number derived from at least one number which has occurred during the computations,

ii) manipulate the data and the pseudo-random number by means of at least one of:

- an arithmetic operation, and
- a logical operation,

so as to obtain a combined set of data, wherein:

- the combined set of data represents an encrypted version of the data in case the computer program is run in encryption mode,

the combined set of data represents a decrypted version of the data in case the computer program is run in decryption mode.

91. A computer being operatively connected to a computer readable data carrier loaded with a computer program for encrypting and decrypting a set of data, the computer program being adapted to run in an encryption mode and in a decryption mode, the computer program being further adapted to:

i) generate a pseudo-random number in a reproducible way by performing the steps of:

- expressing a mathematical system in discrete terms,
- expressing at least one variable of the mathematical system as a fixed-point number,
- performing computations including the at least one variable expressed as a fixed-point number,
- obtaining, from the computations, a resulting number, the resulting number representing at least one of:
  a. a part of a solution to the mathematical system, and
  b. a number usable in further computations involved in the numerical solution of the mathematical system,
- extracting, as the pseudo-random number, a number derived from at least one number which has occurred during the computations,

ii) manipulate the data and the pseudo-random number by means of at least one of:

- an arithmetic operation, and
- a logical operation,

so as to obtain a combined set of data, wherein:

- the combined set of data represents an encrypted version of the data in case the computer program is run in encryption mode,

the combined set of data represents a decrypted version of the data in case the computer program is run in decryption mode,
the computer comprising processor means for running said program.

92. A method of generating a pseudo-random number, the method comprising, in one instance:
I) expressing a mathematical system in discrete terms,
II) defining a seed value representing at least an initial condition for the mathematical system,
III) expressing at least one variable of the mathematical system as a fixed-point number,
IV) performing computations including the at least one variable expressed as a fixed-point number and obtaining a resulting number, the resulting number representing at least one of:
a. a part of a solution to the mathematical system, and
b. a number usable in further computations involved in the numerical solution of the mathematical system,
V) extracting, as the pseudo-random number, a number derived from at least one number which has occurred during the computations,
performing steps I) - V) in a plurality of instances in parallel.

93. A method according to claim 92, comprising transmitting data between the plurality of instances at least while performing step IV) for each of the instances.

94. A method according to claim 92, further comprising transmitting data between the plurality of instances while performing step V) for each of the instances.

95. A method according to claim 92, comprising combining, by use of at least one of:
- an arithmetic operation, and
- a logical operation,

a plurality of pseudo-random numbers extracted at step V) in each of the instances into a common pseudo-random number.

96. A method of performing numerical computations in a mathematical system comprising at least one function, the method comprising the steps of:
- expressing the mathematical system in discrete terms,
- expressing at least one variable of the mathematical system as a fixed-point number,
- performing said computations in such a way that the computations include the at least one variable expressed as a fixed-point number,
- obtaining, from said computations, a resulting number, the resulting number representing at least one of:
  a. at least a part of a solution to the mathematical system, and
  b. a number usable in further computations involved in the numerical solution of the mathematical system,

the step of performing computations comprising:

- repeatedly computing a solution X_{n+1} based on at least one previous solutions X_m, $m \leq n+1$, whereby the step of performing computations is initiated based on at least one initial condition, X_0, of the state variable, X,

the method further comprising:

- providing a cryptographic key as an input to said computations, whereby the cryptographic key is used in generation of the initial condition X_0.

97. A method of determining an identification value for identifying a set of data, the method comprising performing numerical computations in a mathematical system comprising at least one function, the method comprising the steps of:

- expressing the mathematical system in discrete terms,
- expressing at least one variable of the mathematical system as a fixed-point number,
- performing said computations in such a way that the computations include the at least one variable expressed as a fixed-point number,
- obtaining, from said computations, a resulting number, the resulting number representing at least one of:
  a. at least a part of a solution to the mathematical system, and
  b. a number usable in further computations involved in the numerical solution of the mathematical system,

whereby a representation of at least part of the set of data is used in said computations, the method further comprising:

- extracting, as said identification value, at least a part of said resulting number.

98. A method according to claim 97, wherein a cryptographic key is used as a seed value for the computations.

99. A method according to claim 97, wherein the mathematical system comprises at least one of:

- a differential equation,
- a discrete mapping.

100. A method according to claim 99, wherein the differential equation comprises at least one of:

- a partial differential equation,
- an ordinary differential equation.

101. A method according to claim 99, wherein the discrete mapping comprises at least one of:

- an area-preserving map,
- a non area-preserving map.

102. A method according to claim 99, wherein the mathematical system comprises at least one non-linear function governing at least one state variable X.

103. A method according to claim 102, wherein the non-linear mapping function comprises a logistic map of the form $x_{n+1}=\lambda x_n(1-x_n)$, wherein λ is a parameter, x_{n+1} is the value of state variable x at the (n+1)'th stage in the computations, and x_n is the value of state variable x at the n'th stage in the computations.

104. A method according to claim 103, wherein the logistic map is modified into the form $x_{n+1}=\lambda x_n(1-x_n)+\varepsilon(x_n-m_n)$, wherein λ and ε are parameters, x_{n+1} is the value of state variable x at the (n+1)'th stage in the computations, x_n is the value of state variable x at the n'th stage in the computations, and m_n contains a representation of an n'th portion of the set of data.

105. A method according to claim 103, wherein a cryptographic key is used for at least partially determining at least one of the following: λ, ε and an initial value x_0 of state variable x.

106. A method according to claim 97, wherein the mathematical system comprises a set of non-linear mapping functions.

107. A method according to claim 106, wherein the set of mapping functions comprises at least one of:

- an Anosov map of the form:
  $$\begin{bmatrix} x_{n+1} \\ y_{n+1} \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ 1 & 2 \end{bmatrix} \begin{bmatrix} x_n \\ y_n \end{bmatrix} \bmod 1,$$
- a Hénon map of the form:
  $$\begin{bmatrix} x_{n+1} \\ y_{n+1} \end{bmatrix} = \begin{bmatrix} 1 + y_n - a{x_n}^2 \\ bx_n \end{bmatrix}.$$

108. A method according to claim 97, wherein the mathematical system comprises at least one non-linear differential equation.

109. A method according to claim 108, wherein the mathematical system comprises a set of non-linear differential equations.

110. A method according to claim 97, wherein the mathematical system has at least one positive Lyapunov exponent.

111. A method according to claim 97, comprising computing at least one Lyapunov exponent at least once during the mathematical computations.

112. A method according to claim 108, wherein the at least one non-linear differential equation governs at least one state variable, X, which is a function of at least one independent variable, t.

113. A method according to claim 109, wherein the set of non-linear differential equations comprises a Lorenz system.

114. A method of performing numerical computations in a mathematical system comprising at least one function, the method comprising the steps of:

- expressing the mathematical system in discrete terms,
- restricting the range of at least a selected variable of said function, the range being sufficiently narrow so as to exclude values which the selected variable, by virtue of said function, would assume if not restricted by said range,
- performing computations so as to obtain a resulting number, the resulting number representing at least one of:
  a. a part of a solution to the mathematical system, and
  b. a number usable in further computations involved in the numerical solution of the mathematical system,
- when the computations result in a value for the selected variable which is beyond the range, assigning a value within the range to the selected variable.

115. A method according to claim 114, wherein the method is a part of a pseudo-random number generating method.

116. A method according to claim 115, wherein the pseudo-random number generating method generates pseudo-random numbers for use in at least one of encryption and decryption.

117. A method according to claim 114, wherein the mathematical system has at least one positive Lyapunov exponent.

118. A method of performing numerical computations in a mathematical system comprising at least one function, the method comprising the steps of:

- expressing the mathematical system in discrete terms,
- expressing at least one variable of the mathematical system as an integer number,
- placing an imaginary decimal separator in said integer number, whereby the integer number represents a real number,
- performing computations including the at least one variable expressed as an integer number so as to obtain a resulting number, the resulting number being expressed as an integer number,
- positioning the imaginary decimal separator in the resulting number at a predetermined position by performing at least one of the steps of:
  - correcting the position of the imaginary decimal separator in the integer number, and
  - placing an imaginary separator in the resulting number.

119. A method of performing numerical computations in a mathematical system comprising at least one function, the method comprising the steps of:

- expressing the mathematical system in discrete terms,
- expressing at least one variable of the mathematical system as a fixed-point number,
- performing said computations in such a way that the computations include the at least one variable expressed as a fixed-point number,
- obtaining, from said computations, a resulting number, the resulting number representing at least one of:

  a. at least a part of a solution to the mathematical system, and

  b. a number usable in further computations involved in the numerical solution of the mathematical system.